

2A by the box labeled “Configuration Version Data” accompanying the publisher configuration 220 in FIGS. 2A and 2B.

[0041] Two types of assemblies are possible, those having strong names, e.g., including a public key signature or the like such that any two different assemblies can be unambiguously identified, and those having a simple name, which may be ambiguous in the system, and, for example, do not contain a public key. To provide isolation, any simply-named assemblies on which an application manifest specifies a dependency are treated as privatized assemblies 210 of FIG. 2A, (wherein the dashed box represents that the privatized assemblies may or may not be present for a given application). Privatized assemblies are those that the application does not intend to be shared with other applications. Privatized assemblies that have simple names are normally installed into the same folder as the executable application code, whereby the assembly is isolated and the application is given the effect of being monolithic, e.g., its assemblies are not influenced by different assemblies having the same simple name, and so forth. By not being shared, privatized assemblies with simple names thus have the benefit of virtually complete isolation, at the expense of the benefits obtained by sharing. Assemblies having strong names do not need to be placed into the application folder for isolation purposes, because assemblies with strong names are known to be the exact one the application needs. Thus, instead of being privatized by storing in a certain directory, such assemblies may be placed into a global assembly cache 212 to obtain the benefits of sharing, while effectively preserving isolation, because an application that asks for a strongly named assembly will get a copy that is exact (subject to configuration overrides, as described below).

[0042] The application, manifest and assemblies, both privatized and global, are installed to their appropriate file system locations at the time the application or assembly is installed. In general, the application, application manifest and privatized assemblies (those not strongly named) are copied to the application folder, while strongly named assemblies may be copied to the global assembly cache 212 (e.g., one or more folders). Note that to provide side-by-side existence of assembly versions, any existing assembly versions are not overwritten in the assembly cache 212 when

another version is installed, (although a version can be removed by other means, at the risk of breaking an application that depends on that version). The assembly cache can be hidden and/or access controlled to prevent assemblies from being easily removed. The installation and general usage of manifests and assemblies are further described in the aforementioned United States Patent Application entitled “Isolating Assembly Versions for Binding to Application Programs.”

[0043] Example manifests in XML format are set forth in the tables below, wherein TABLE1 is an example of a simple application manifest where the application depends on a side-by-side version of COMCTL32:

TABLE 1

<pre>&lt;?xml version="1.0" encoding="UTF-8" standalone="yes"?&gt; &lt;assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0"&gt;   &lt;assemblyIdentity     version="1.0.0.0"     processorArchitecture="X86"     name="Microsoft.Windows.mysampleApp"     type="win32"   /&gt;   &lt;description&gt;Your app description here&lt;/description&gt;   &lt;dependency&gt;     &lt;dependentAssembly&gt;       &lt;assemblyIdentity         type="win32"         name="Microsoft.Windows.Common-Controls"         version="6.0.0.0"         processorArchitecture="X86"         publicKeyToken="6595b64144ccf1df"         language="*"       /&gt;     &lt;/dependentAssembly&gt;   &lt;/dependency&gt; &lt;/assembly&gt;</pre>
--

[0044] TABLE2 is an example of an application manifest where the application depends on a side-by-side version of COMCTL32 and an assembly is privatized to the application:

TABLE 2

<pre>&lt;?xml version="1.0" encoding="UTF-8" standalone="yes"?&gt; &lt;assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0"&gt;   &lt;assemblyIdentity     version="1.0.0.0"     processorArchitecture="X86"     name="Microsoft.Windows.mysampleApp"     type="win32"   /&gt;   &lt;description&gt;Your app description here&lt;/description&gt;   &lt;dependency&gt;     &lt;dependentAssembly&gt;       &lt;assemblyIdentity         type="win32"         name="Microsoft.Windows.Common-Controls"         version="6.0.0.0"         processorArchitecture="X86"         publicKeyToken="6595b64144ccf1df"         language="*"       /&gt;     &lt;/dependentAssembly&gt;   &lt;/dependency&gt; &lt;/assembly&gt;</pre>
--